

Remarks/Arguments

Claims 1-42 are pending in the application. Claims 1, 21 and 42 are independent.

There are no claim amendments.

Claim 1 recites: *a method for providing dynamic interaction between a pair of application programs by an interface module of a terminal, the pair of applications including a requestor application desiring access to a target application, the method comprising the steps of:*

registering access information of the target application, the access information including published access information made available in a data structure for retrieval by the interface module;

receiving an access request by the interface module from the requestor application, the access request including request content corresponding to the published access information of the target application;

obtaining an interface component by using the request content to search the data structure, the interface component configured for enabling communication between the interface module and the target application in an access format expected by the target application; and

employing the interface component by the interface module to satisfy the access request of the requestor application for interaction with the target application.

The Examiner has rejected claims 1-7, 12-14, 19-27, 32-34 and 39-42 under 35 U.S.C. 102(e) as being anticipated by Slaughter, US patent no. 7,458,082. Applicant respectfully traverses the rejections.

As described in the Background of the application as filed, the problem the subject invention addresses relates to communication between related applications. Specifically, if an application interface changes, it is also required to change many, or all, of the related or dependent applications to maintain compatibility and interoperability.

The solution as disclosed provides an interface module that facilitates the communication between a requestor application and a target application. The target application registers access information, such as an application name and its corresponding parameters. The requestor application submits an access request for the target application, which is received by the interface module. The requestor application also provides content corresponding to the published access information of the target application.

The interface module uses the received information (in a generic format such as XML, for example) to obtain an interface component, which is configured to enable communication between the interface module and an access format expected by the target application. The interface component is then used by the interface module to satisfy the access request.

In effect, the interface module acts as an intermediary agent on behalf of the requestor application to facilitate communication. In this way, requestor applications only have to know how to communicate with the interface module. The interface module can then communicate with the target application via the interface component. Accordingly, if the target application changes, all that needs to be changed is the interface component and, possibly, the published access information of the target application..

In contrast, the invention of Slaughter is characterized in Col. 1 of Slaughter thus:

1. Field of the Invention

This invention relates to distributed computing environments including Web-centric and Internet-centric distributed computing environments, and more particularly to bridging a heterogeneous distributed computing environment based upon a message passing model for connecting network clients and services to other environments based upon other models. 30

Further motivation behind the invention of Slaughter is revealed at Col. 2, and then at Col. 6 respectively as follows:

What is needed is a simple way to connect various types of intelligent devices to allow for communication and sharing of resources while avoiding the interoperability and complex configuration problems existing in conventional networks. Various technologies exist for improving the addition of devices to a network. For example, many modern I/O buses, 5

It may be desirable for such clients to have a mechanism for finding and invoking distributed applications or services. The client may need to be able to run even large legacy applications which could not possibly fit in the client's memory footprint. As discussed above, current technology, such as 65

There are a number of elements of claim 1 that are not present in Slaughter.

Firstly, for example, the interface module is equated by the Examiner with the API layer 102. However, there is no teaching of the API layer 102 retrieving published access information of a target application, or of a desired service in the case of Slaughter. Nor is there any teaching of the API layer 102 receiving an access request from the client including request content corresponding to published access information .

Secondly, Slaughter does not disclose "obtaining an interface component by using the request content to search the data structure, the interface component configured for enabling communication between the interface module and the target application in an access format expected by the target application". Slaughter discloses a point-to-point communication between a client and an end

service via API layer 102, with discovery of the end service facilitated using existing device discovery protocols (Col. 78, Ln. 58-60) so there is no need for *“obtaining an interface component by using the request content to search the data structure, the interface component configured for enabling communication between the interface module and the target application in an access format expected by the target application”*.

Thirdly, although the Examiner stated that the claim 1 element *“employing the interface component by the interface module to satisfy the access request of the requestor application for interaction with the target application”* is inherent in Slaughter, Applicant respectfully disagrees with this conclusion. Since an interface component is not present in Slaughter, for the reason discussed immediately above, it cannot be necessary to Slaughter. The Examiner’s conclusion is in error, as a matter of law, since the interface component is not necessary to the functioning of the invention of Slaughter. To support such conclusion of inherency, the Examiner must prove necessity. If the element is not present as a necessity, then that element is not inherent, as a matter of law. *MEHL/Bioplille Int’l Corp. v. Milgraum*, 192 F.3d. 1362, 1365, 52 USPQ2d 1303, 1305 (Fed. Cir. 1999).

Therefore, for at least the reasons discussed above, Applicant submits claim 1 is patentable in view of Slaughter and, as such, requests that the rejection of claim 1 be withdrawn.

Independent claims 21 and 42 include similar limitations as claim 1, and therefore a corresponding argument applies. Accordingly, Applicant submits that the rejection to these claims be withdrawn for at least the same reasons discussed above with regard to the Slaughter reference.

Since the remaining dependent claims depend from one of the above noted independent claims, Applicant submits that the rejection of these claims be withdrawn for at least the same reasons.

For the foregoing reasons, the Applicant respectfully submits that the claimed invention is patentable over the prior art. Reconsideration and allowance are respectfully requested.

Respectfully submitted,

/Henry Ohab/

Henry Ohab
Registration No. 45,854
416 862 3593

Gowling Lafleur Henderson LLP
1 First Canadian Place, Suite 1600
Toronto, Ontario, M5X 1G5

TOR_LAW\ 7200015\1